



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/998,848	11/15/2001	Kenneth Y. Ogami	CYPR-CD01177M	6884

7590 01/26/2006
WAGNER, MURABITO & HAO LLP
Two North Market Street, Third Floor
San Jose, CA 95113

EXAMINER

VO, TED T

ART UNIT PAPER NUMBER

2191

DATE MAILED: 01/26/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/998,848	Applicant(s) OGAMI, KENNETH Y.	
	Examiner Ted T. Vo	Art Unit 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 07 October 2005.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-14 and 16-35 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-14 and 16-35 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 3/14/05 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

Art Unit: 2191

DETAILED ACTION

1. This action is in response to the amendment filed on under Request for Continued Examination on 10/07/2005.

Claims 1-14, 16-35 are pending in this application.

Response to Arguments

2. Applicants' arguments in the Remarks section filed on 10/07/2005 have been fully considered.

In the Applicants' arguments,

Applicants alleged:

"Applicant respectfully agrees that Bindra does not address automatically constructing source code," as admitted on page 5 of the Office Action mailed June 14, 2005. The IDE Tutorial was cited in that Office Action to correct this deficiency. However, the IDE Tutorial does not qualify as prior art, a fact admitted to in the instant Advisory Action."

Examiner responds: In an argument filed by Applicants on 3/14/05, Applicants alleged, "Applicant respectfully submits that the qualified references cited in the Office Action do not show or suggest "automatically constructing source code," in particular source code "comprising configuration information for a programmable block of said microcontroller corresponding to said virtual block" and "wherein said configuration information is used to cause said programmable block to implement said function" as recited in independent Claim 1. Claims 2-12 are dependent on Claim 1 and recite additional limitations." (Emphasis added).

Responsive to this argument, Examiner noted that Bindra does not explicitly address "automatically constructing source code". However, the examiner' position is that, even there are no "exactly statements" as claimed, in the Bindra, the PSoC Designer does have "automatically constructing source code".

The prior art, "The Tutor" is added to prove that the "PSoC Designer" has such features.

The added "The Tutor" cannot be disqualified art because it simply admits a feature that belongs in a tool existing and used by the Public more than a year before these filing claims. The fact is that the added art mentions and teaches developers how to use the "PSoC Designer". The

Art Unit: 2191

clause "PSoC Designer" is the fact for showing the Bindra and the Tutor are discussing to the same tool.

It should be noted that when a reference like Bindra describing a tool like PSoC Designer, it does not need to detail everything in the tool. The Bindra has established the Examiner's position that there is a tool, "PSoC Design" that has been known and used by the Public more than a year before the Claims seeking for a protection. Bindra clearly suggests that by entering configuring information into programmable block as shown in Figure 4, the editor provides mapping the configuring information (of a predetermined programmable block) into pre-built program code or user modules. Finally, the developer only connects the pins to form a SoCBlock on-chip (Note: It assumes that Applicants' argument filed by Applicants on 3/14/05 traversed this feature that qualified references cited in the Office Action do not show or suggest "automatically constructing source code."). This has been addressed in the bottom of Figure 4. The buttons shown on top of the editor is simply common used, as seen in other tools, for mapping the configuration and programmable functional block to a source file.

In the similar manner, the specification's drawings (Figures 6-7) are identical to the PSoC Design [Device Editor] (shown on top of the Drawing Figures). In order to support the claimed feature above, the specification (p.19: 1-5) shows that

Parameter window 510 displays additional user-specifiable information about the selected user module 500. It is appreciated that this information is used for parameterization of the user module. After assigning virtual blocks to all selected user modules, the user then presses a generate code button 507 to generate files in step 470. (Emphasis added)

The "PSoC Definer" [Device Editor] clearly includes the claimed feature. However, using the added reference, "The Tutor" it aims the Examiner's position that the "PSoC Definer" [Device Editor] has the claimed feature, and this feature had been used/known by the Public more than a year before the filing.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

A person shall be entitled to a patent unless –

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claim 1-14, 16-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bindra, "Programmable SoC Delivers A New Level Of System Flexibility", 2000, in view of Tutorial Revision 1.0 (hereinafter: Tutor), "PSoC Designer: Integrated Development Environment", 7-2001.

Given the broadest reasonable interpretation of followed claims in light of the specification.

As per Claim 1:

Bindra discloses a PSoC Designer that is used to configure and construct code for a microcontroller; the disclosure covers the limitations,

A method for configuring a microcontroller, comprising:

displaying a collection of virtual blocks in a design system with each virtual block in said collection corresponding to a programmable block in said microcontroller (Bindra: See P.11, Figure 4);

receiving a selection of a user module defining a function (Bindra: See Figure 4: "User Modules Selected for Placement": E.g. see circuit block icons in the right top section which will be implemented in a combination shown within the right bottom section in the Figure 4);

Art Unit: 2191

assigning a virtual block taken from said collection to said user module (See Figure 4, Each block in the circuit in the right bottom section in Figure 4 could be assigned in this section from selection of "User Modules" in the right top section; configuration information and connection are assigned by dialog section in the left section and the buttons given in the top rows of the PSoC Designer Tool); *automatically constructing source code comprising configuration information for a programmable block of said microcontroller corresponding to said virtual block wherein said configuration information is used to cause said programmable block to implement said function* (Bindra: For this limitation, see Figure 4 and its below illustration: "which are next mapped onto the SoCblocks on-chip". For *configuration information*: refer to "Global Resources" and "Placement parameters" in the left section of Figure 4).

Bindra does not explicitly address the claimed statement, "*automatically constructing source code*" (as applicants' argument filed on 3/14/05).

The Tutor, "PSoC Designer: Integrated Development Environment" shows that the PSoC Designer has the feature, "*automatically constructing source code comprising configuration information*" (Tutor: See whole page 14) for the circuit in Figure 4 of Bindra. Particularly, in Tutor, page 14, see text pointed by the symbol '=' and see Figure 13, and refer "The Application code has been generated successfully'. For *configuration information*: refer to Figures 7-9 in pages 10-11.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to know that the PSoC Designer discussed by Bindra includes the tutoring feature "*automatically constructing source code comprising configuration information*" of Tutor, as conforming to an included and available feature of the PSoC Designer [Device Editor].

As per Claim 2: Bindra further discloses,

The method of Claim 1, wherein said function comprises a pulse width modulator (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 36, "PWMs").

Art Unit: 2191

As per Claim 3: Bindra further discloses, *The method of Claim 1, wherein said function comprises a timer.* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 36, "timers").

As per Claim 4: Bindra further discloses, *The method of Claim 1, wherein said function comprises an analog-to-digital converter* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 35, "ADCs").

As per Claim 5: Bindra further discloses, *The method of Claim 1, wherein said function comprises a digital-to-analog converter* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 35 "DACs").

As per Claim 6: Bindra further discloses, *The method of Claim 1, wherein said function comprises a counter* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 36 "counters").

As per Claim 7: Bindra further discloses, *The method of Claim 1, wherein said function comprises a signal amplifier.* (See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 33 "differential amplifiers").

As per Claim 8: Bindra further discloses, *The method of Claim 1, wherein said function provides serial communication.* (See Figure 4, refer to "User Module" that represents various Digital functions, and see P.3, line 9, "serial transmitters/receivers").

As per Claim 9: Bindra further discloses, *The method of Claim 1, wherein said collection is displayed as a two dimensional array of programmable analog virtual blocks and programmable digital virtual blocks.* (See collections in the right bottom section, which is two dimensional array).

As per Claim 10: Bindra further discloses, *The method of Claim 1, wherein said assigning further comprises assigning a second virtual block to said user module* (See collections in the right bottom section, which is two dimensional array).

As per Claim 11: Bindra further discloses, *The method of Claim 1, wherein said source code comprises a symbolic name for a register address in said programmable block.* (Bindra: See page 2, lines 12-17 ('register space that holds the configuration information').

Art Unit: 2191

As per Claim 12: Bindra further discloses, *The method of Claim 11 wherein said symbolic name is derived from said function.* (See Bindra 'User module' in Figure 4, where user module represents a circuit element. Each circuit element is a symbolic name function: e.g.: ADC, DAC, Timer, Counter, etc).

As per Claim 13: Bindra disclosure covers the limitations,

A method of configuring a microcontroller having a programmable block, said method comprising: receiving a selection of a user module defining a circuit design (Bindra: See Figure 4: "User Modules Selected for Placement": E.g. see circuit block icons in the right top section which will be implemented in a combination shown within the right bottom section in the Figure 4); assigning a virtual block in a design system where said virtual block corresponds to said programmable block (See Figure 4, Each block in the circuit in the right bottom section in Figure 4 could be assigned in this section from selection of "User Modules" in the right top section; configuration information and connection are assigned by dialog section in the left section and the buttons given in the top rows of the PSoC Designer Tool); and automatically constructing assembly code comprising configuration information for said programmable block to implement said circuit design, wherein said assembly code is constructed from template assembly code by substituting information specific to said user module and information specific to said circuit design for generic information in said template assembly code (Bindra: For this limitation, see Figure 4 and its illustration below the Figure, "which are next mapped onto the SoCblocks on-chip". For configuration information: refer to "Global Resources" and "Placement parameters" in the left section of Figure 4).

Bindra does not explicitly address the claimed statement, *"automatically constructing assembly code comprising configuration information"* (as applicants' argument filed on 3/14/05).

The Tutor, "PSoC Designer: Integrated Development Environment" shows that the PSoC Designer has the feature, *"automatically constructing assembly code comprising configuration information"* (Tutor: See whole page 14; particularly see first full paragraph, "existing assembly-

Art Unit: 2191

source": Claimed limitation: *from template assembly code*) for the circuit in Figure 4 of Bindra. Particularly, in Tutor, page 14, see text pointed by the symbol '=>' and see Figure 13, and refer "The Application code has been generated successfully". For *configuration information*: refer to Figures 7-9 in pages 10-11 (Tutor).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to know that the PSoC Designer discussed by Bindra includes the tutoring feature "*automatically constructing source code comprising configuration information*" of Tutor, as conforming to an included and available feature of the PSoC Designer [Device Editor].

As per Claim 14: Bindra in view of Tutor further discloses,

"The method of Claim 13, wherein said automatically constructing further comprises:

computing a register address for a register within said programmable block; determining a symbolic name for said register address, said symbolic name corresponding to said user module and said circuit design; and substituting said symbolic name for a generic name in said template assembly code". See page 2, lines 12-17 ('register space that holds the configuration information') and page 6, lines 7- 13, ('user modules are selected, pins are assigned, and register mapping are establish') (Further refer to Tutor page 14 for *automatically constructing*) for cover the limitations:

- *computing a register address for a register within said programmable block*: referring "register mapping"
- *determining a symbolic name for said register address, said symbolic name corresponding to said user module and said circuit design*: referring "holds the configuration information".
- *substituting said symbolic name for a generic name in said template assembly code*: referring the code construction performed by the PSoC Designer shown by the Tutor.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include because the Tool, PSoC Designer, used in the Tutor and the Tool, PSoC Designer, disclosed by Bindra are the same.

As per Claim 16: Bindra in view of Tutor further discloses the limitations of Claim 16.

Art Unit: 2191

See page 2, lines 12-17 ('register space that holds the configuration information') and page 6, lines 7- 13, ('user modules are selected, pins are assigned, and register mapping are establish') (Further refer to Tutor page 14 for *automatically constructing*).

- *determining a symbolic name corresponding to said user module and said circuit design;*

referring "holds the configuration information".

- *computing a register address for a register within said programmable block; referring "register mapping"*

- *assigning said symbolic name to said register address; and placing said symbolic name into said assembly code in place of a generic name provided in said template assembly code file:* referring the code construction performed by the PSoC Designer.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include because the Tool PSoC Designer used in the Tutor and the Tool PSoC Designer disclosed by Bindra are the same.

As per Claim 17: Bindra disclosure covers the limitations,

A method of configuring a microcontroller having a programmable block, said method comprising:

receiving a selection of a user module defining a function; (Bindra: See Figure 4: "User Modules Selected for Placement": E.g. see circuit block icons in the right top section which will be implemented in a combination shown within the right bottom section in the Figure 4);

assigning a virtual block in a design system where said virtual block corresponds to said programmable block (See Figure 4, Each block in the circuit in the right bottom section in Figure 4 could be assigned in this section from selection of "User Modules" in the right top section; configuration information and connection are assigned by dialog section in the left section and the buttons given in the top rows of the PSoC Designer Tool); and

automatically constructing assembly code with personalization information specifying said programmable block as performing said function, wherein said assembly code is constructed from template assembly code by substituting information specific to said user module and information specific to said function for generic information in said template assembly code (Bindra: For this

Art Unit: 2191

limitation, see Figure 4 with its illustration below, "which are next mapped onto the SoCblocks on-chip". For *personalization information*: refer to "Global Resources" and "Placement parameters" in the left section of Figure 4).

Bindra does not explicitly address the claimed statement, "*automatically constructing assembly code with personalization information specifying said programmable block as performing said function*" (as applicants' argument filed on 3/14/05).

The Tutor, "PSoC Designer: Integrated Development Environment" shows that the PSoC Designer has the feature, "*automatically constructing assembly code with personalization information specifying said programmable block as performing said function*" (Tutor: See whole page 14; particularly see first full paragraph, "existing assembly-source": Claimed limitation: *from template assembly code*) for the circuit in Figure 4 of Bindra. Particularly, in Tutor, page 14, see text pointed by the symbol '=>' and see Figure 13, and refer "The Application code has been generated successfully'. For *personalization information*: refer to Figures 7-9 in pages 10-11 (Tutor).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to know that the PSoC Designer discussed by Bindra includes the tutoring feature, "*automatically constructing assembly code with personalization information*" of Tutor, as conforming to an included and available feature of the PSoC Designer [Device Editor]. As per Claim 18: Bindra in view of Tutor further discloses,

"The method of Claim 17, wherein said automatically constructing further comprises:
computing a register address for a register within said programmable block;
determining a symbolic name for said register address, said symbolic name corresponding to said user module and said function; and
placing said symbolic name into said assembly code".

See page 2, lines 12-17 ('register space that holds the configuration information') and page 6, lines 7- 13, ('user modules are selected, pins are assigned, and register mapping are establish') (Further refer to Tutor page 14 for *automatically constructing*) for cover the limitations:

Art Unit: 2191

- *computing a register address for a register within said programmable block:* referring "register mapping"
- *determining a symbolic name for said register address, said symbolic name corresponding to said user module and said function:* referring "holds the configuration information".
- *placing said symbolic name into said assembly code:* referring the code construction performed by the PSoC Designer shown by the Tutor.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include because the Tool PSoC Designer used in the Tutor and the Tool PSoC Designer disclosed by Bindra are the same.

As per Claim 19: Regarding: *substituting said symbolic name in place of a generic name provided in said e template assembly code*, by referring the code construction performed by the PSoC Designer shown by the Tutor.

As per Claim 20: Bindra in view of Tutor further discloses the limitations of Claim 20.

See page 2, lines 12-17 ('register space that holds the configuration information') and page 6, lines 7- 13, ('user modules are selected, pins are assigned, and register mapping are establish') (Further refer to Tutor page 14 for *automatically constructing*).

- *determining a symbolic name corresponding to said user module and said function:* referring "holds the configuration information".
- *computing a register address for a register within said programmable block:* referring "register mapping".

assigning said symbolic name to said register address; and placing said symbolic name into said assembly code: referring the code construction performed by the PSoC Designer.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include because the Tool PSoC Designer used in the Tutor and the Tool PSoC Designer disclosed by Bindra are the same.

As per Claim 21: Regarding,

"A method of configuring a microcontroller having a programmable block, said method comprising:

Art Unit: 2191

*receiving a selection of a user module defining a function having a control parameter;
assigning a virtual block in a design system where said virtual block corresponds to said
programmable block; and
constructing assembly code for operating said control parameter within said programmable block,
wherein said assembly code is constructed from template assembly code by substituting
information specific to said user module, information specific to said function and information
specific to said control parameter for generic information in said template assembly code".*

Claim has the functionality corresponding to the functionality of Claim 17. See rationale addressed in the rejection of Claim 17 above.

As per Claim 22: Regarding, *"The method of Claim 21, wherein said constructing further comprises:*

*computing a register address for a register within said programmable block;
determining a symbolic name for said register address, said symbolic name corresponding to said
user module and said function; and placing said symbolic name into said assembly code".*

Claim has the functionality corresponding to the functionality of Claim 18. See rationale addressed in the rejection of Claim 18 above.

As per Claim 23: Regarding, *"The method of Claim 22, wherein said placing further comprises:
substituting said symbolic name in place of a generic name provided in said template assembly
code".*

Claim has the functionality corresponding to the functionality of Claim 19. See rationale addressed in the rejection of Claim 19 above.

As per Claim 24: Regarding, *"The method of Claim 21, wherein said constructing further comprises:*

*determining a symbolic name corresponding to said user module and said function;
computing a register address for a register within said programmable block;
assigning said symbolic name to said register address; and
placing said symbolic name into said assembly code".*

Art Unit: 2191

Claim has the functionality corresponding to the functionality of Claim 20. See rationale addressed in the rejection of Claim 20 above.

As per Claim 25: Regarding,

"A method of configuring a microcontroller having a programmable block, said method comprising:

receiving a selection of a user module defining a function having a control parameter;

assigning a virtual block in a design system where said virtual block corresponds to said programmable block;

constructing an assembly code routine using said control parameter, wherein said assembly code routine is constructed from template assembly code by substituting information specific to said user module, information specific to said function and information specific to said control parameter for generic information in said template assembly code; and constructing a header file referencing said assembly (Tutor: page 14: "project library source, PSoCConfig.asm" and/or "Application Program Interface", which are/is combined when generating Application files) code routine".

Claim has the functionality corresponding to the functionality of Claim 17. See rationale addressed in the rejection of Claim 17 above.

As per Claim 26:

Claim has the functionality corresponding to the functionality of Claim 1. See rationale addressed in the rejection of Claim 1 above.

As per Claim 27: Bindra further discloses, "two dimension array" (collection shown the right bottom section of Figure 4).

As per Claim 28: Bindra further discloses, "assigning a second virtual block" (which is either one of other blocks shown the right bottom section of Figure 4).

As per Claim 29: Bindra further discloses "assembly code" which is the code generated by the PSoC Design to the collection shown in the right bottom section of Figure 4, where the symbolic name for a register address is done by register mapping as addressed above.

Art Unit: 2191

As per Claim 30: Bindra further discloses, "derived from said function" which is based on the icon, "user module" which is functionalized to a circuit element, and based on pins assignment to the user module.

As per Claim 31: Bindra disclosure covers the limitation:

A computer implemented method of generating program information for a programmable electronic device, said method comprising:

a) accessing a selected a user module, wherein said user module is defined by a first data structure; (Bindra: See Figure 4: "User Modules Selected for Placement"- E.g. see circuit block icons in the right top section implemented in a combination shown within the right bottom section in the Figure 4);

b) placing said user module within a second data structure that defines a hardware resource of said programmable electronic device; (See Figure 4, Each block in the circuit in the right bottom section in Figure 4, could be assigned in this section from selection of " User Modules" in the right top section; configuration information and connection are assigned by dialog texts in the left section buttons on the top of the PSoC Designer Tool. When running the Tool, each user module assigned in the collection in the right bottom section will be assigned accordingly);

"c) using said first and second data structures to automatically generate first source code for realizing said user module within said hardware resource; and

d) saving said first source code in a computer file"

(Bindra: For this limitation, see Figure 4 and its below illustration: "which are next mapped onto the SoCblocks on-chip", '*using said first and second data structures*'. For saving *said first source code in a computer file*: refer to 'File', 'Edit', 'View' on the top of the PSoC Designer Tool).

Bindra does not explicitly address the claimed statement, "*automatically generate first source code for realizing said user module within said hardware resource*".

The Tutor, "PSoC Designer: Integrated Development Environment" teaches "*automatically generate first source code for realizing said user module within said hardware*

Art Unit: 2191

resource" (Tutor: See whole page 14) for the circuit in Figure 4 of Bindra. Particularly, in Tutor, page 14, see text pointed by the symbol ' \Rightarrow ' and see Figure 13, and refer "The Application code has been generated successfully".

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to know that the PSoC Designer discussed by Bindra includes the tutoring feature "*automatically generate first source code for realizing said user module within said hardware resource*" of Tutor, as conforming to an included and available feature of the PSoC Designer [Device Editor].

As per Claim 32: Bindra in view of Tutor further discloses,

A method as described in Claim 31 further comprising:

e) accessing parameter values that define the behavior of said user module such that it operates in a prescribed manner, (Bindra: See left section in Figure 4);

f) automatically generating second source code, based on said parameter values, for causing said user module of said hardware resource to behave in said prescribed manner (See Tutor and its text discussed in page 14); *and*

g) saving said second source code in a computer file (Bindra: See Figure 4, icons in the top rows used to save a file).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine because the Tool PSoC Designer used in the Tutor and the Tool PSoC Designer disclosed by Bindra are the same.

As per Claim 33: Bindra further discloses, "*A method as described in Claim 32 further comprising using said first and second source code to program said programmable electronic device*" because it the collection in Figure 4 would be mapped to a real design.

As per Claim 34: Bindra further discloses, *A method as described in Claim 33 wherein said programmable electronic device is a microcontroller.* See Bindra's Figure 1.

As per Claim 35: Bindra in view of Tutor further discloses, *A method as described in Claim 31 wherein said a) and said e) are performed using a graphical user interface,* because PSoC Designer is a GUI.

Art Unit: 2191

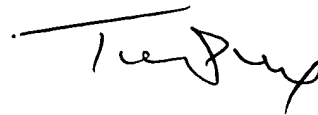
Conclusion

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ted T. Vo whose telephone number is (571) 272-3706. The examiner can normally be reached on 8:00AM to 5:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y. Zhen can be reached on (571) 272-3708.

The facsimile number for the organization where this application or proceeding is assigned is the Central Facsimile number **571-273-8300**.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



Ted T. Vo
Primary Examiner
Art Unit 2191
January 20, 2006